

# 8月25日笔记\_stream\_mybatis Criteria

笔记:

## java stream:

stream的使用方法

jdk8

方法名	方法作用	方法种类	是否支持链式调用
count	统计个数	终结	否
forEach	逐一处理	终结	否
filter	过滤	函数拼接	是
limit	取用前几个	函数拼接	是
skip	跳过前几个	函数拼接	是
map	映射	函数拼接	是
concat	组合	函数拼接	是

- 终结方法：返回值类型不再是 `Stream` 接口自身类型的方法，因此不再支持类似 `StringBuilder` 那样的链式调用。本小节中，终结方法包括 `count` 和 `forEach` 方法。
- 非终结方法：返回值类型仍然是 `Stream` 接口自身类型的方法，因此支持链式调用。（除了终结方法外，其余方法均为非终结方法。

类似sql的使用：

```

list

    .stream

    .filter(e -> e.d == 2)

    .map(e -> {

        Class AB = new Class();

        AB.a = e.a;

        AB.b = e.b;

        return AB;

    })

    .collect(Collectors.toList());

```

stream与其他集合的转化

```

List<String> list = new ArrayList<>();

// ...

Stream<String> stream1 = list.stream();

Set<String> set = new HashSet<>();

// ...

Stream<String> stream2 = set.stream();

Vector<String> vector = new Vector<>();

// ...

/// map
Stream<String> stream3 = vector.stream();

Stream<String> keyStream = map.keySet().stream();

Stream<String> valueStream = map.values().stream();

Stream<Map.Entry<String, String>> entryStream = map.entrySet().stream();

```

## Stream性能:

在少数据量的处理场景中 (size <= 1000) stream 的处理效率是不如传统的 iterator 外部迭代器处理速度快的, 但是实际上这些处理任务本身运行时间都低于毫秒, 这点效率的差距对普通业务几乎没有影响, 反而 stream 可以使得代码更加简洁;

在大量数据 (size > 10000) stream 的处理效率会高于 iterator, 特别是使用了并行流, 在 cpu 恰好将线程分配到多个核心的条件下 (当然 parallel stream 底层使用的是 JVM 的 ForkJoinPool, 这东西分配线程本身就很玄学), 可以达到一个很高的运行效率, 然而实际普通业务一般不会有需要迭代高于 10000 次的计算;

Parallel Stream (并行流) Parallel Stream 受引 CPU 环境影响很大, 当没分配到多个 cpu 核心时, 加上引用 forkJoinPool 的开销, 运行效率可能还不如普通的 Stream;

使用建议 简单的迭代逻辑, 可以直接使用 iterator, 对于有多步处理的迭代逻辑, 可以使用 stream, 损失一点几乎没有的效率, 换来代码的高可读性是值得的;

单核 cpu 环境, 不推荐使用 parallel stream, 在多核 cpu 且有大数据量的条件下, 推荐使用 parallel stream;

stream 中含有装箱类型, 在进行中间操作之前, 最好转成对应的数值流, 减少由于频繁的拆箱、装箱造成的性能损失。

性能测试: <https://www.cnblogs.com/ffpy/p/15715802.html>

总体上, 常规迭代循环, 性能更高一点, 数据量越大, 差异更明显。

## mybatis Example[Criteria]:

mybatis-generator会为每个字段产生如上的Criterion, 如果表的字段比较多,产生的Example类会十分庞大。理论上通过example类可以构造你想到的任何筛选条件。在mybatis-generator中加以配置, 配置数据表的生成操作就可以自动生成example了。具体配置可以参考MBG有关配置。下面是mybatis自动生成example的使用。

创建方法:

```
Criteria criteria = session.createCriteria(Mytest.class);
criteria.orEqualTo();// 或操作
criteria.andEqualTo();// 与操作
criteria.addLike();// 模糊操作
```